

# Open Research Online

---

The Open University's repository of research publications and other research outputs

## Decentralized Learning Infrastructures for Community Knowledge Building

### Journal Item

#### How to cite:

De Lange, Pater; Göschlberger, Bernhard; Farrell, Tracie; Neumann, Alexander Tobias and Klamma, Ralf (2019). Decentralized Learning Infrastructures for Community Knowledge Building. IEEE Transactions on Learning Technologies (Early Access).

For guidance on citations see [FAQs](#).

© 2019 IEEE

Version: Accepted Manuscript

Link(s) to article on publisher's website:

<http://dx.doi.org/doi:10.1109/TLT.2019.2963384>

---

Copyright and Moral Rights for the articles on this site are retained by the individual authors and/or other copyright owners. For more information on Open Research Online's data [policy](#) on reuse of materials please consult the policies page.

---

[oro.open.ac.uk](http://oro.open.ac.uk)

# Decentralized Learning Infrastructures for Community Knowledge Building

Peter de Lange, Bernhard Göschlberger, Tracie Farrell, Alexander Neumann and Ralf Klamma

**Abstract**—Learning in Communities of Practice (CoPs) makes up a significant portion of today’s knowledge gain. However, only little technological support is tailored specifically towards CoPs and their particular strengths and challenges. Even worse, CoPs often do not possess the resources to host or develop a software ecosystem to support their activities. In this contribution, we describe a decentralized learning infrastructure for community knowledge building. It takes into account the constant change of these communities by providing a lightweight and scalable infrastructure, without the need for central coordination or facilitation. As a real use case, we implement a question-based dialog application for inquiry-based learning and ignorance modeling with our infrastructure. Additionally, we explore the possibility of using social bots to connect the services provided by the decentralized infrastructure to communication tools already present in most communities (e.g. chat platforms). Following a design science approach, we describe a multi-step evaluation of both the infrastructure and application, together with the improvements made to the resulting artifacts of each step. Our results indicate the relevance of our approach, that may serve as an example of how decentralized learning infrastructures for learning outside of formal settings can be applied by CoPs for knowledge building.

**Index Terms**—learning infrastructures, knowledge building, communities of practice, design science, non-formal learning

## I. INTRODUCTION

THE vast majority of human learning happens outside of formal settings. Learning activities may be quite informal, as found in incidental learning, self-regulated learning and socialization [1]. Some learning may involve more structure or planning, which is generally referred to as non-formal learning [2]. A significant portion of this learning happens in Communities of Practice (CoPs) [3]. These communities are not bound together by an organization, but rather by sharing a common craft or profession, with the desire to learn from each other through knowledge sharing and knowledge building. While only few CoPs have the size and influence to get tools tailored to their needs, the long tail [4] of CoPs does not possess the resources, such as central hosting infrastructures or shared budget. Consequently, they often adopt publicly available tools (e.g. social software) and re-purpose them according to their needs, mitigating the tools’ technical shortcomings through socially enforced usage policies. These

(mostly unwritten) policies include the knowledge necessary to navigate within the digital community space and are an entry barrier for novices as well as a hindrance to community coherence. Moreover, the CoP becomes dependent on the tool provider and also loses control over its data. Even if a CoP manages to establish a centralized infrastructure, this often results in dependencies on single, knowledgeable members or institutions and does not account for dynamic membership, a common characteristic of CoPs.

As a consequence, we claim that a suitable infrastructure for CoPs needs to be decentralized and managed by the community members themselves. It should be easily deployable, extensible and flexible in terms of scalability and accessibility from the outside. Finally, it should also provide support for orientation and self-organization within the community’s digital space. The microservice paradigm [5], with loosely coupled services, bound together by lightweight protocols, fits these demands perfectly. Combined with an underlying peer-to-peer (p2p) network of nodes managed by the CoPs themselves, the microservices should self-replicate through the network according to the community’s current needs and provide the necessary information. Once deployed on the infrastructure, those services and development efforts should remain available, even after the contributing member has left the CoP. Like the ship in the Theseus paradox, a community should be able to persist, even though all of its members have changed over time, as long as there are people willing to engage. Serving as a *community’s long term memory*, the infrastructure allows members to learn from their “ancestors”, much like we can observe in scientific communities.

Just like opening the water tap, using a certain learning environment should be available to every community member at all times. This formulates also the requirement, that the learning environment is easily accessible to non-technical community members. We introduce the utilization of the conversational interfaces that social bots offer, so members can connect to their learning environment in ways already familiar to them. Thus, we propose a *Learning as a Utility* approach, which makes it possible for all community members to equally engage in development, hosting and using learning applications.

The contribution of this work is twofold. First, we describe a decentralized infrastructure that provides CoPs with an independent, sustainable and flexible way of developing, hosting and sharing their state-of-the-art learning applications on the Web. Second, we present a digitized and distributed, version of a proven method for inquiry-based learning and knowledge building [6], built with the decentralized infrastructure.

P. de Lange, A. Neumann and R. Klamma are with the Department of Databases and Information Systems, RWTH Aachen University, Germany (e-mail: {lastname}@dbis.rwth-aachen.de).

B. Göschlberger is with both the Research Studios Austria FG, Austria and the Johannes Kepler University Linz, Austria (e-mail: goeschlberger@researchstudio.at).

T. Farrell is with the Open University UK, United Kingdom (e-mail: tracie.farrell@open.ac.uk).

We start by presenting the background of our research and continue with a real-world use case, from which we derive our functional requirements (Sec. II). After a short functionality description of the developed application (Sec. III), we present our design science-based research methodology (Sec. IV). Next follows a detailed description of the artifacts developed in this contribution (Sec. V). We evaluate our artifacts in multiple iterations and discuss their implications (Sec. VI), before presenting related work (Sec. VII) and concluding this contribution (Sec. VIII).

## II. BACKGROUND AND USE CASE

### A. Ignorance Modeling in Communities of Practice

Our work focuses on the support of community learning processes in the digital space. We understand it as a social process that involves negotiation of meaning and social construction of knowledge. With respect to learning as a social process, the theory of CoPs describes the emergence, transfer and preservation of knowledge [3].

In the domain of school education, a theory that specifically focuses on social configuration for knowledge creation is the theory of *knowledge building* by Scardamalia and Bereiter [7]. The rationale behind it is that the knowledge called “state-of-the-art” is the sum of the knowledge of the community. Knowledge work therefore is the advancement of the state of knowledge within a CoP. Knowledge building explicitly focuses on the community knowledge advancement and stresses the temporary nature of ideas and theories. Every idea is improvable and every theory can be refined, redefined or replaced by a new improved theory. To work on ideas, knowledge building uses a form of discourse that can be characterized as a cooperative process where participants are committed to

- 1) progress,
- 2) seek common understanding,
- 3) and expand the base of accepted facts.

Knowledge building assumes that learners’ understanding is emergent and that the development of complex cognitive structures for complex concepts is achieved by self-organization: “new conceptual structures [...] emerge through the interaction of simpler elements [...]” [7]. This is also applicable to *knowledge of ignorance*, which can rather be expressed by questions than by idea statements.

Coming from the field of organizational studies and knowledge management, the *SECI model* developed by Nonaka and Takeuchi [8] and its adaption to Web 2.0 [9] describe the process of knowledge creation in four cyclic steps:

- 1) **Socialization** (tacit to tacit): the process of sharing tacit knowledge by collaboration and practice, through which learners develop a shared mental model
- 2) **Externalization** (tacit to explicit): make this knowledge explicit, e.g., by writing it up, revealing the tacit knowledge
- 3) **Combination** (explicit to explicit): combine explicit knowledge sources to create new knowledge
- 4) **Internalization** (explicit to tacit): by using the explicit knowledge sources, the knowledge is internalized

For emergent knowledge, revealed ignorance plays a pivotal role in both the theory of knowledge building [10] and the SECI model (here especially in the “externalization” step, where both knowledge and ignorance can be revealed). The learning process of *Inquiry-Based Learning* (IBL) starts with a question or statement of curiosity, sometimes called the “wonder moment” [11]. Once an unanswered question is asked within a community, it challenges the ideas and theories of the community. A collective model of *community ignorance* results from the subsequent discourse.

### B. Use Case: European Youth Workers

In our use case, a community of young European youth workers are preparing for participation in a European-funded training course on “creative leadership”. The participants are an international group, with different levels of experience, from multiple organizations and countries. While they may not yet constitute a CoP, these young adults form a Community of Inquiry (CoI) as a precursor to identifying areas of shared practice [12], eventually leading to a CoP. The trainer team must create learning content that appeals to this diverse group and meets their needs, which is a challenge, given the complexity of both creativity and leadership as learning subjects. In addition, the three trainers providing the course are distributed across different countries and organizations as well, with no possibility to meet beforehand. Since the whole CoP neither shares a geographic location, nor central infrastructure or budget, this use case stands exemplary for the needs and challenges of distributed CoPs.

To help establish the boundaries of the participants’ knowledge and identify common ground or potential conflicts, the trainers want to find out which questions the participants have about creative leadership and how those questions relate to one another. Specifically, the trainers implement a form of *Question-Based Dialog* called Noracle [6] before the training starts, to model and visually represent their common space of ignorance about creative leadership. This special form of IBL starts with a seed question raised by the trainers, which is then answered by the participants by raising follow-up questions. This way, the *Community Ignorance* becomes visible and the trainers gain insight about what the participants are interested in and their views on the subject. As participants create this *Problem Space*, they document the questions they have about creative leadership, their assessments of the questions that others stated and any links they perceive between them. In its analog form, this involves an on-scene session at the start of the training course, where the community has a limited time-frame to establish their community ignorance by writing down questions they have. A digital version of the concept, hosted decentrally by the community itself, could be applied already before the community meets. We state the following two research questions:

- R1: Does a digital version affect the community’s knowledge of their ignorance?
- R2: Can a decentralized learning infrastructure be managed by the community?

### III. DIGITAL QUESTION-BASED DIALOG FOR MODELING IGNORANCE

In this section, we describe the functionality of a digital and distributed version of the Noracle method. It fulfills the use case described in the previous section and makes it possible to explore and map community ignorance through question-based dialog, asynchronously and without a formal infrastructure.

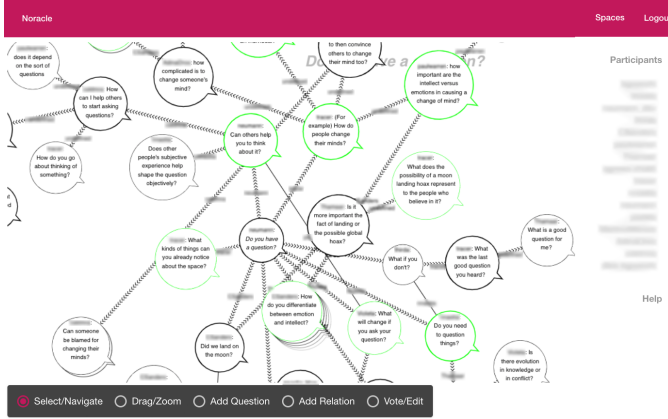


Fig. 1. Screenshot of the Distributed Noracle application, showing a question-based dialog space used in one of our evaluations.

A space is the main view of the application (shown in Fig. 1). Users can create a space and invite others to the space by sharing an invitation link. The user interface provides a list of subscribed spaces such that users can switch between spaces with two clicks. The space view consists of a canvas displaying the questions and their relations as a graph of speech bubbles. It also features a list of users subscribed to the space and a (collapsible) help section. Below the canvas, users can select their current interaction mode. The “Select/Navigate” mode allows users to define the portion of the graph that is displayed. Selected questions and direct neighbors of selected questions are displayed. If a displayed question that is not yet selected has neighbors that would be displayed upon selecting it, they are symbolically indicated as additional speech bubbles behind the question. In the “Drag/Zoom” mode, users can move questions around freely, as well as pan and zoom, to either view parts of the graph in detail or get a birds eye view. The “Add Question” and “Add Relation” mode allows users to add questions or relations by clicking on one question (add a question) or two questions (add a relation). Then, a dialog window opens that asks the user to enter the text of the question or the type of the relation. For relations, we allow for both *Follow Up* relations (depicted as small arrows indicating the direction), which is the default type of relation that is created between a new question and its parent question, as well as *Link* relations (depicted as straight lines) that display a certain connection of similar questions, although they are not in a direct *Follow Up* relationship. Finally, the “Vote/Edit” mode enables users to either modify their own questions and relations or to assess the value of questions or relations of others. We use a coloring mechanism that displays the entity according to its overall rated usefulness in a specific color, ranging from green to red.

### IV. METHODOLOGY

Our methodology follows a design science approach as proposed by Hevner [13], and applies the guidelines proposed by Peffers [14]. Fig. 2 gives an overview on the whole process, consisting of seven iterations. While Sec. VI provides a detailed description of each evaluation step and its outcome, this section describes the overarching process.

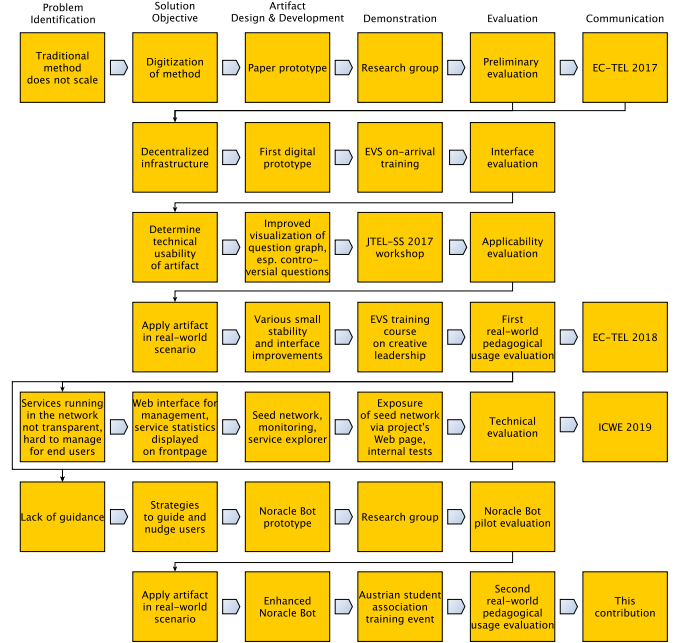


Fig. 2. The design science process we followed to design, develop, evaluate and communicate our decentralized learning infrastructures for community knowledge building.

Our starting point was the original, analog Noracle method [6] and its problem of scalability. The preliminary evaluation, based on a paper prototype, led to the requirement of the decentralized infrastructure. We communicated these results in a vision paper [15]. Our next phase was mainly concerned with getting to know how people would interact with our newly developed prototype and the interface evaluation describes the first evaluation of the digital artifact. We continued with a first evaluation of the decentralized scenario in a workshop setting, which disclosed technical shortcomings we tried to overcome and improve for the next phase, the first real-world pedagogical usage evaluation of our artifact. This rather large evaluation allowed us insights into manifold aspects of both infrastructure and tool usage. The aggregated results of these three iterations were communicated in [16]. Based on the outcomes of this first real-world evaluation, we found several technical shortcomings of our approach that we addressed in the following iteration. We established the seed network, improved the monitoring facilities and developed the service explorer, which we evaluated in our technical evaluation. These results were communicated in [17]. The lack of guidance, especially with regards to larger question-based dialog spaces was addressed in the following iteration by introducing the Noracle Bot, which we describe in our pilot

bot evaluation and finally our second real-world pedagogical usage evaluation. The complete results of the whole process are communicated here.

## V. A DECENTRALIZED LEARNING INFRASTRUCTURE

In the following, we first present an overview of our technical infrastructure, before we describe the realization of the Distributed Noracle in more detail. We start with introducing an exemplary usage scenario in Sec. V-A, before we introduce the underlying p2p basis for distributing the communities' learning infrastructure in Sec. V-B. Sec. V-C provides an overview on the service explorer that is used by community members to start and stop services. We continue with Sec. V-D by presenting the social bot integration we use in our later evaluations to guide users through the question-based dialog. Finally, Sec. V-E describes the realization of the Distributed Noracle with the help of these components.

### A. Exemplary Usage Scenario

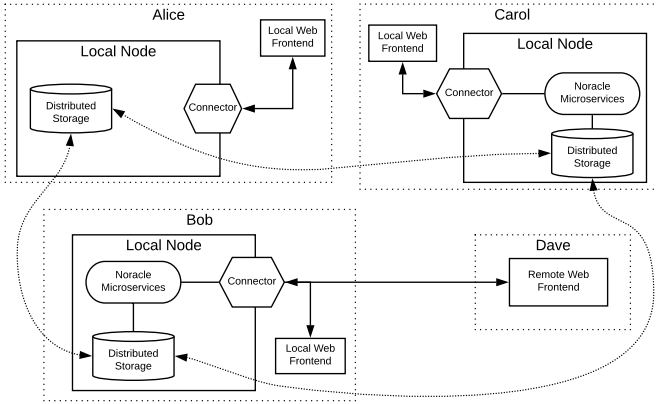


Fig. 3. Exemplary usage scenario of the Distributed Noracle.

Fig. 3 shows an exemplary usage scenario of a Distributed Noracle session. While *Bob*'s node features the set of microservices that realize the application (see Sec. V-E), *Alice* has decided to start an empty node without any services running on it. This can have several reasons, including the lack of resources, both in terms of computing power or, especially in mobile settings, energy. *Carol*'s node also contains a set of Noracle microservices, whilst *Dave* has not started a node at all and uses *Bob*'s node to access the remote Web frontend for participating in the collaborative session. As this scenario demonstrates, our framework provides flexible access to the application with several possibilities to join a session. Depending on the currently available resources of a community member, our framework allows to flexibly start and stop (parts of) applications on a node. This usage scenario does not feature any centralized component, like a master node or a central URL for the Web frontend. Rather, the whole infrastructure is distributed among the community.

### B. Technical Basis

The technical basis we use for this work is called *las2peer* [18], an open source p2p framework for implementing

and hosting Java microservices. Every *las2peer* node in our decentralized community learning infrastructure consists of at least two components. The first is the *Distributed Storage*. This storage is partitioned and partly duplicated throughout the network, allowing for a shared, yet synchronized data store. Technically, we base our storage and inter-node communication mechanisms on *Pastry* [19], a p2p overlay network that provides both a messaging system as well as a *DHT* (Distributed Hash Table) storage system. To ensure privacy, security and data protection, we added end-to-end encryption in form of an *Envelope* system on top of it, ensuring each message and all data stored on the infrastructure is encrypted. The second component a node has to integrate is the so-called *RESTful Web Connector*. It realizes the communication to the outside, with the capability of routing *RESTful* [20] calls to an application's (gateway) interface.

Our framework is capable of load balancing requests to microservices in the entire network, may it be because the service simply does not exist on the local node, or the node is currently overloaded with requests and offloads the task to other nodes in the network. Upstarting services register themselves to the network by calling a specific routine of the node, which then manages their location in the distributed storage for all nodes to look-up. This *Sidecar Pattern*-like [5] service registration and discovery ensures that a connector will find the nearest service that currently is flagged as being capable of taking requests. Additionally, a blockchain-based decentralized service registry keeps record of all running services at all time (cf. Sec. V-C).

The communication between microservices is realized using a *Message Oriented Middleware* (MOM) [21] that is based on the *Publish & Subscribe Pattern* [22]. Each node registers all running services as subscribers to their corresponding "Service Topic". If a service wants to call another service, it performs a remote method invocation that is sent throughout the network. A node hosting a corresponding service that receives this request will route it to the service, which will handle it. The answer is then sent again in the same way throughout the network. Several timeout mechanisms and an acknowledgment system prevent messages with missing receiver to be forwarded endlessly or messages being answered by multiple services. By using the p2p network to enforce an *Event-Driven Architecture* (EDA) of microservice-based applications [23], we target the needs of fast-changing topologies in CoPs, where complete knowledge of the network might both not be available or even desirable. Nodes can join and leave the network at any time, and the network keeps a persistent distributed storage with *Eventual Consistency* (following the BASE model of modern cloud computing architectures [24]), regardless of the current topology. Besides this, it is of course possible for a microservice to implement and maintain its own database, separately of the distributed storage.

### C. The Service Explorer

In a more recent addition to the framework, we implemented a decentralized service registry and discovery mechanism [17], targeted at both end-users and developers, based on blockchain

technology [25]. The requirements of this arose from the use case described in Sec. II-B, and particularly surfaced in the first pedagogical usage evaluation described later on in this contribution. Here, we had to rely on a slightly artificial network setup (see Sec. VI-D for more details) due to technical shortcomings of our framework in both controlling which services are available in a network, as well as how to start them from a non-technical user perspective.

The service registry enables both end-users and developers to easily find service releases, verify their origin and either use remote instances or replicate the service to their own node. Although most of these requirements could be solved by using some kind of central service registry, this approach has one major drawback: it redirects the power over the infrastructure from the community to the maintainer of this centralized component and thus contradicts the whole idea of decentralization. Without the ability to authorize service releases, the community relies on the service registry to forward their discovery requests, which raises the same issues a decentralized infrastructure tries to tackle. To be in line with the concept and preserve its advantages, las2peer's decentralized service registry is governed by the whole community in terms of authorizing service releases and validating service instances. Combining the completeness and time-preserving properties of a blockchain with the space-efficiency of the DHT-based distributed storage allows us to utilize the strengths of each technology and compensate their respective weaknesses.

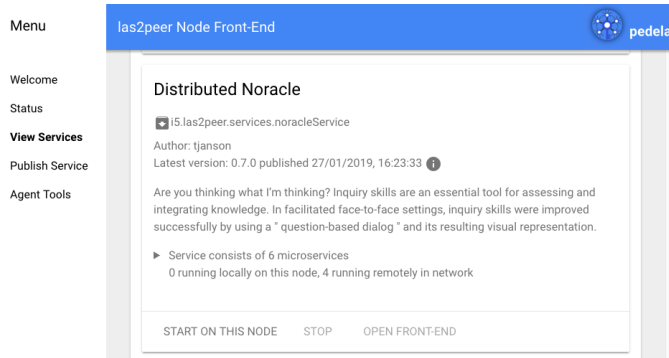


Fig. 4. Screenshot of the las2peer service explorer, currently displaying the Distributed Noracle application.

From an end-user's point of view, the outcome of this work is the service explorer, depicted in Fig. 4. This particular example shows that the Distributed Noracle application is only partly deployed in the network (four of six microservices running remotely in the network), while none of the services are deployed on the node the user is accessing. This information comes directly from the private blockchain that we host in parallel to the network. The user can now decide to either start the two remaining services on her node or start all of the services that realize the application locally.

#### D. Integration of Social Bots

As our evaluations grew larger, also did the resulting question-based dialog spaces. We identified the need for more assistance for users of the tool to navigate their way through

the spaces. Thus, the most recent addition to the framework and also the Distributed Noracle application is the integration of a social bot that is capable of sending messages via a chat interface to users, informing them of recent changes to the graph, and possibly interesting areas worth exploring. Here we make use of the concept of nudging [26], by pointing users to areas in the graph relevant to them, encouraging them to produce content and also to provide relevant information to facilitate reflection.

In this contribution, we use Slack<sup>1</sup> as the conversational interface, because of its widely spread use in professional communities. The messages are sent daily and provide information about the community's activity in the Distributed Noracle within the last 24 hours. All questions mentioned in the messages are provided as links directly to the corresponding Distributed Noracle space, with only the linked question initially selected, such that the user starts exploring the graph from this question when clicking on a link in the bot message. Fig. 5 is an example of the general statistics that the bot sends to a public channel, to be seen by all participants. It starts

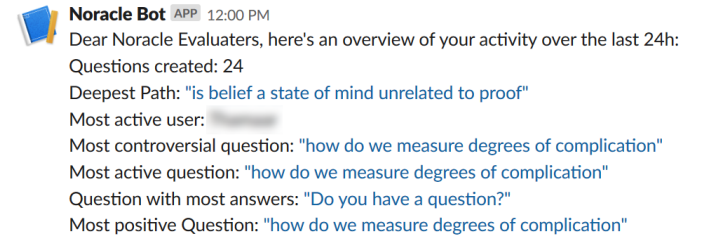


Fig. 5. An exemplary general bot message, as it was sent to the evaluators during our pilot bot evaluation (cf. Sec. VI-G).

with the number of questions created, followed by the question with the deepest path, the question that is most distant from the seed question of the space. It is followed by the most active user. The activity includes the creation of follow-up questions, relations and rating questions. The next link directs to the most controversial question, which is the question with the most votes in both directions (helpful/not helpful). Similar to the most active user, the message also provides the most active question which caused the most follow-ups relations and votes. Finally, the question that caused the most follow-up questions and the question with the most positive feedback are presented to the community.

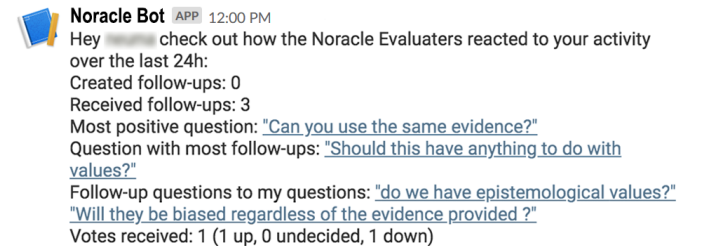


Fig. 6. A personal bot message send to an evaluator during our pilot bot evaluation (cf. Sec. VI-G).

<sup>1</sup><https://slack.com>

Fig. 6 shows a personalized message, send to a specific user as a private message. The message gives information about the number of follow-up questions the participant has created and how many follow-up questions the participant's questions have received. In addition, the question that got the most positive feedback, the question that raised most follow-up questions and each individual follow-up question of the user's questions are displayed. Finally, the number of votes received is provided.

### E. Building the Distributed Noracle

The Distributed Noracle application consists of a set of five microservices that realize different functionalities of the application, and a gateway service to route incoming requests. A *Space Service* handles the creation of spaces and their members. The *Question Service* takes care of creating and updating questions, while the *Relation Service* does the same for relations. The *Vote Service* handles votes for both questions and relations. Finally, the *Agent Metadata Service* is responsible for storing additional metadata (such as the name) for the members of the CoP. Additionally to these five services, the *Noracle Service* serves as the *Gateway Service* to the application that provides a RESTful API to the outside. Being called by the connector, it distributes the requests to the set of microservices we just described. The frontend of our application is based on the Angular framework and it is part of the node, served from the distributed storage. Therefore, we developed a *File Service* that provides a RESTful interface for storing and serving Web frontends directly from the network, removing the need for an additional Web server. Authentication is done using the OpenId Connect *Single Sign-on* (SSO) standard.

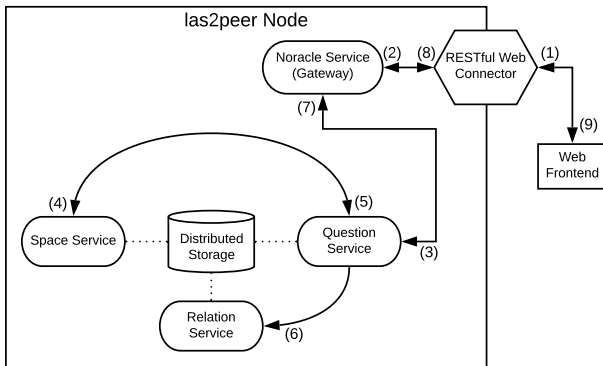


Fig. 7. A question creation process in the Distributed Noracle.

To give a concrete example of inter-microservice communication, consider an incoming request for creating a question (see also Fig. 7). This RESTful request is transferred from the *RESTful Web Connector* (1) to the *Noracle Service* (2), which sends a request to the *Question Service* (3). This service in turn invokes the corresponding *Space Service* (4) for further details, for example if the user is allowed to create a question in this particular space. Upon receiving the answer from the

*Space Service* (5), the *Question Service* creates a new *Question* object in the distributed storage and calls the *Relation Service* (6) for creating the corresponding relation between the newly created question and its parent. Finally, the *Question Service* answers to the *Noracle Service* (7), which sends the reply to the *RESTful Web Connector* (8), who forwards the HTTP Response to the *Web Frontend* (9), whether the question has been successfully created.

This particular scenario is not necessarily limited to a single node, the microservices can be situated anywhere in the network and it is also neither needed nor desired that a particular microservice knows which instance of the called microservice did handle the request. In the exemplary usage scenario depicted in Fig. 3, if Alice's node receives such a request, it is distributed throughout the network, because Alice's node does not host any of the application's microservices. Depending on their current load, the request would be processed by the node of either Carol or Bob, and their *Noracle Service* would possibly distribute the just described sub-request again to microservices on other nodes. The flexible scalability of the infrastructure also allows several instances of the same microservice residing at a node, spawning automatically according to the current need. The infrastructure is designed for failure in such a way, that non-responding microservices are automatically shut-down and replaced by new instances.

To provide CoP members with the software needed to start their own node, we created a *Node Package*. It is a small folder that contains an empty node preconfigured to connect to a network via a (configurable) *Seed Node*. It then replicates the microservices of the application via the p2p network and starts them locally. For requirement analysis and feedback, we used the Requirements Bazaar [27] to also include end-users in improving the development of the application and underlying framework itself. Both application and framework are released as open source software<sup>2</sup>.

## VI. EVALUATION

We evaluated our application in multiple iterations, with different types of learning communities. Following the design science methodology described in Sec. IV, each evaluation had a certain focus that lead to a gradual improvement of our approach and implementation. In the following, we present these evaluations and finish this section with a discussion of the outcomes.

### A. Preliminary Evaluation

In the preliminary evaluation, a Web science research group used a paper mock-up of the Distributed Noracle for questioning current priorities in their research field. The purpose of this evaluation was to determine whether the method could be transferred to a digital space and which features would be required. This community was appropriate because of the shared interest in a topic, diverse levels of experience, and a loose collaborative structure.

<sup>2</sup><https://distributed-noracle.github.io>



**Participants and Procedure:** 8 members of the community took part in the trial. Half of the participants were more experienced members of the team, as determined by whether or not they were supervising PhD students. The other half were PhD candidates or post-doctoral researchers. To represent a shared digital space, the participants worked asynchronously on a large poster in the lab. A general reflection question was posed as the seed question in the Distributed Noracle mock-up: “What is the most relevant, open question for social semantics?” Each participant received a differently colored marker to represent her contributions to the poster. As participants added questions, they were also asked to circle questions they supported and draw links between questions to show their relationship. Participants also starred those contributions they thought were most helpful. The evaluation lasted for three days.

**Analysis and Outcomes:** After concluding the exercise, the participants took part in a short group discussion regarding the insights they could draw from looking at the question graph. The main themes of the discussion were:

- 1) the tool could help to structure dialog more efficiently
- 2) it encourages to consider broader or new perspectives
- 3) participants need assistance in interpreting the graph

The participants also expressed thoughts about the overall value of the proposed artifact. They emphasized the additional possibilities a digital version would provide in terms of longer running efforts to structure their thoughts as a group. The need to transfer the process of question-based dialog to a digital space to increase its value was established through this evaluation.

### B. Interface Evaluation

The first evaluation of the digital tool was conducted with participants on an “on arrival” training for participation in the European Voluntary Service (EVS) program. The participants used the Distributed Noracle to consider the future of European youth work in the context of a project planning session. This community was appropriate because of the ill-defined nature of the topics that participants were exploring and their lack of having a central infrastructure.

**Participants and Procedure:** 7 participants between the age of 20-25 from different European and Erasmus+ partner countries took part in the study. The participants had similar levels of experience in the area of youth work (1-2 years). In this evaluation, the participants worked synchronously. All participants used a given link to access a single-node deployment of the Distributed Noracle. After a project planning session in their face-to-face seminar, the participants joined the space and continued their reflections online. They had a set period of time to explore the application with the general reflection question posed to them “What is the future of European Youth Work?”. As participants added questions, they were also asked to assess questions they found helpful and create links between different questions to show their relationship. The exercise lasted for approximately 30 minutes, followed by semi-structured interviews with the participants regarding usability and value of the digital tool.

**Analysis and Outcomes:** Since this was the first evaluation of the digital version we focused on the aspects that inherently differed from the face-to-face version. In particular we looked at analytic features designed to help the individual to get a sense of a question’s *importance, quality and validity* for the group. Examples for this are the marking of questions where conflicts are present in red, or darkening the circle that surrounds the topic as more and more contributors agree that the question is relevant. All participants agreed that they understood the semantics of these analytic features. Some participants suggested improvements regarding the visual representation. The most frequently mentioned suggestions for improvements concerned the layout and animations of the graph itself. Some participants considered the automatic force-directed layouting system slightly disorientating at times. We attributed this to the prototypical nature of the design artifact in this first evaluation and improved the overall look and feel in further iterations.

### C. Applicability Evaluation

The applicability evaluation was conducted with workshop participants of the Joint European Summer School on Technology Enhanced Learning (JTELSS). The purpose of this evaluation was to test the technical features of the tool, in particular the decentralized architecture and its applicability within a group of people with diverse technical and non-technical backgrounds. The community was considered appropriate for a this kind of evaluation because of their experience with educational software.

**Participants and Procedure:** Approximately 20 people attended the workshop. First, the participants were given a short introduction to the method of question-based dialog and to the application. As part of this introduction, participants were instructed on how to start their own node and join the network. Participants used their own devices to launch their nodes. We provided a local seed node that participants could connect to. The participants were then given about 20 minutes of time to explore the tool. We provided a general starting question in a sample space. Participants were also asked to assess questions they found helpful and create links between different questions to show their relationship. In addition, they were invited to create their own space and invite other participants to join. In the end, we asked participants to fill an online questionnaire.

**Analysis and Outcomes:** We received 7 questionnaire responses. Participants shared the same conceptual understanding about the possible usage context of the tool: 5 responses were similarly themed around conceptual mapping, development of a common understanding and knowledge expansion. The visual representation played a significant role: 4 participants expressed the importance of question-color and -size. Regarding usage monitoring, we were able to capture data from 12 workshop participants, of which 8 participants started their own node that connected to our seed node. This network of nodes also provided access to the application for the other 4 participants, who were unable to launch an own node on their device. They used the Web frontend provided



by another participant's node (cf. usage scenario *Dave* in Fig. 3). This demonstrated the capability of our approach to overcome technical hurdles, such as firewall restrictions or device security policies. The data we received from this evaluation was afterwards used to improve the application, leading to a more stable version used in our first pedagogical usage evaluation.

#### D. First Real-World Pedagogical Usage Evaluation

The first real-world pedagogical usage evaluation was conducted with the community described in the initial use case (cf. Sec. II-B). Participants of an European training course on creative leadership were invited to take part in an experiment using the Distributed Noracle to help prepare for the course and explore their existing knowledge gaps about the topic. For the organizational team hosting the training course, an orientation activity of this kind is conducted typically at the beginning of a course. Depending on the methods used and the complexity of the course, this can take several hours to achieve with participants. The incentive for the host organization to take part was in improving the on-boarding process for participants and gaining an initial understanding of their ideas. The purpose of this evaluation was to test the application in a real asynchronous and distributed setting, adding monitoring data to the qualitative verbal and written data.

**Participants and Procedure:** 34 participants took part in the evaluation. The participant group was diverse, with different nationalities, levels of experience and knowledge about the subject of the training course, *Creative Leadership*. One week before the training course, participants were notified via email that an "experiment" would be taking place, using a beta version of an application to help prepare for the training. They were informed that their participation in the experiment was completely voluntary, but that the training team and researchers felt the tool could be helpful in establishing what this particular group of participants found most confusing or difficult about the concept of creative leadership. The participants received information on how to join the Distributed Noracle and were invited to contribute their own questions to a specific reflection question ("What is creative leadership?").

Since the participants were locally distributed with prior contact only via email, we created an artificial distributed setting by creating a network of nodes at a university. We provided a URL to the participants that automatically distributed them to their specific node. This created a scenario where each participant had her own node, without the actual need for a technical setup procedure that would have been unfeasible for this particular evaluation, especially regarding the evaluation of the results. After the first 48 hours, participants were asked via email to review the questions that other participants had posted so far and evaluate how important or useful they are to the overall discussion. Participants were also encouraged to make links between questions and add relevant follow-up questions to the questions of other participants. Once the participants arrived at the training course, the entire trainer team and the trainees participated in an analysis of the question graph and an evaluation of the tool's features. The evaluation

included three items: *What insights can you draw from the graph? What features or functions might improve the value of this tool for you? In which situations could you imagine to use it?* Each individual had five minutes to review the graph and to take some notes. Then, the facilitator gathered the insights in a plenary session, during which the participants' statements were also clustered according to their shared theme.

**Analysis and Outcomes:** With regard to the insights that could be drawn from the graph, the participants found it quite easy to see what was most important to the group, such as focusing on the development of *creative skills*. When asked how they evidence this with the graph, the participants noted that many questions related to this topic in some way. The graph also showed a considerable agreement about the importance of questions related to this topic (as indicated by the green color).

Participants stated, that the graph helped them to realize they had taken a *very individualistic perspective on creativity and leadership*, with very few questions having to do with social aspects of creative development. Considering that a large part of the training course was founded on shared leadership and joint creativity, this was important to the training team to have highlighted in the graph. The way that questions were formulated allowed the participants to differentiate between questions related to defining creativity ("what" questions) and questions related to the process of developing or improving creativity ("how" questions).

The trainees agreed that the tool helped establish the interests of a group in advance. This is useful in a variety of settings, in particular educational settings that are blended or fully online. They also felt that participating in the experiment was a valuable use of their time. Using the tool in this way saved the training team an estimated three hours of time with the participants on the training course, allowing them to more quickly engage with the subject.

The training team remarked, that instructions were important in helping the participants to know how to use the application. Especially with new users, they recommend facilitation to maintain the quality of the space by demonstrating question-asking and some of the application's additional features. Features that participants felt were important to develop had to do with analytics to help uncover other types of insights or consequences. For example, only one trainee had noticed that similar questions were repeated several times in the graph. When the training team highlighted this point to the group, they agreed that this was valuable information that they missed as the graph became larger and less visually manageable. In addition, a third of the participants said that they would find it helpful if there was a way of knowing exactly how many people or a percentage of people found a question useful. Participants felt this information could help them identify questions they might find interesting or important. All of the participants and the trainer team felt that the tool would be improved by having a way of visualizing what insights or consequences could be drawn, for quick and easy reference.

Beginning with this evaluation, we also began live monitoring the complete network for user activities [28]. We started the monitoring the day we sent out the invitation mail, while

we asked the participants to start their 48h collaboration phase on the beginning of day three. With the help of the monitoring data, we recorded high activity between day three and five, while it declined afterwards. Still, the number of recorded activity before and after this “official” trial phase shows the *intrinsic motivation* participants had to (re)visit the problem space, an important factor for learning activities in self-regulated learning scenarios. Another interesting observation we made during analyzing the monitoring data was, that with an average question depth of 1.9, a question was on average about two questions away from the seed question. We perceive this as another indicator of the usefulness of the graph-based visualization, since most questions did not connect directly to the seed question, but to follow-up questions, demonstrating the evolving awareness of the community ignorance, represented by the growth of the graph.

After this evaluation, we implemented the service registry with its service explorer frontend (Sec. V-C) to allow non-technical community members easier handling of services and applications via a node-frontend, while also verifying services running in the network and detecting the upload of malicious services to the network. We then continued with a technical evaluation to measure capabilities of our framework and application in terms of larger spaces and high usage simulations.

#### E. Technical Evaluation

Apart from our user evaluations, we also evaluated the Distributed Noracle application from a technical point of view. This work was mainly done in between the first pedagogical usage evaluation and the integration and evaluation of the Noracle Bot, but we continued to monitor usage activity thenceforward.

**Procedure:** From the first pedagogical usage evaluation on, we maintained a network of ten always-online “seed” nodes. This served two purposes. On the one hand, these nodes acted as an entry point to the Distributed Noracle application directly by providing multiple Web connector endpoints that are displayed on our project’s Web page. On the other hand, they allowed to connect one’s own node to the network to participate. Additionally to these nodes, we installed a “monitoring node” that collected (anonymized) usage statistics of the Distributed Noracle. The pedagogical implications of this monitoring data already were partly covered in Sec. VI-D. But from there we went one step further and added pre-processed monitoring data to the landing page of the application’s frontend. This way, each user that either started her own local frontend, or used one of the provided ones, was able to see them.

Additionally to this, we wanted to test the technical capabilities of our application and framework in terms of data processing and provision. Therefore, we took a snapshot of the data of a Reddit megathread<sup>3</sup> and processed it in a Distributed Noracle space, measuring the time it took. It consisted of 163 users, posting 308 questions and answers. We interpreted

#### Usage Statistics

Number of Questions: 626  
Number of Spaces: 38  
Average Questions per Space: 16.47  
Average Longest Thread per Space: 3.23

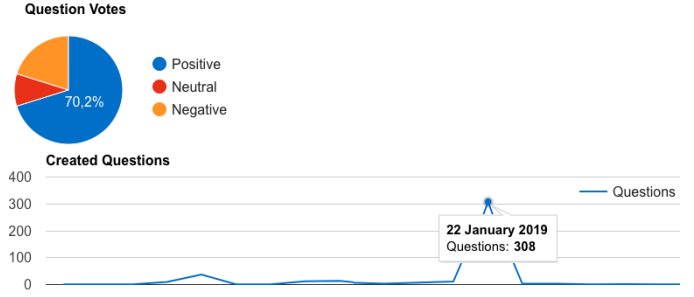


Fig. 8. Usage statistics of the one-year decentralized infrastructure deployment, continuously providing the Distributed Noracle application since our first real-world pedagogical usage evaluation.

each response, regardless of its type (question or answer), as a follow up question to its parent.

**Analysis and Outcomes:** Fig. 8 shows a screenshot of the usage statistics collected in a roughly one-year period since the start of the first pedagogical usage evaluation. Clearly visible is the peak of created questions on the day we uploaded the Reddit data to the network, making up for about half of the 626 created questions in this time period. Interesting observations are that the majority of people rather rate questions as helpful (positive) than neutral or not helpful (negative). The average (median) number of questions per space is quite low with only 16.5 questions stated per space. We interpreted this as a clue that guidance and nudging support, as we introduce later with the addition of the Noracle Bot, are crucially needed to handle larger spaces.

Regarding the Reddit megathread, the time it took to create all 163 users was about 3 minutes, whilst the creation of the 308 questions took about 100 minutes. It has to be noted, that the latter time is also the result of not being able to parallelize the process, since a question could only be uploaded once its parent question existed, so that it could be linked to it. Nevertheless, in a real use case of such a big thread, multiple questions could be created in parallel, since many questions have the same parent. This evaluation showed that the framework is technically capable of processing large question-based dialog spaces. However, Fig. 9 shows a (partial) screenshot of the resulting graph. As one can see, the space is very crowded. Apart from the more-or-less obvious seed question in the center of the space, it would be very hard for users to find relevant areas in the graph. Also, the visualization tended to re-render pretty often due to the large amount of questions to be displayed on the screen. Especially in mobile usage scenarios, a large space like this would become impractical to use.

On the basis of the now available technical evaluation data and the newly implemented monitoring capabilities, we thought about ways of providing relevant analytical information about the space to the community. The approach that we felt made most sense to focus on, given the pedagogical and technological underpinnings of the Distributed Noracle, as well as the improvements suggested by participants, was

<sup>3</sup>[https://www.reddit.com/r/NoStupidQuestions/comments/addb81/us\\_government\\_shutdown\\_megathread/](https://www.reddit.com/r/NoStupidQuestions/comments/addb81/us_government_shutdown_megathread/) (Snapshot taken on 22.01.2019)

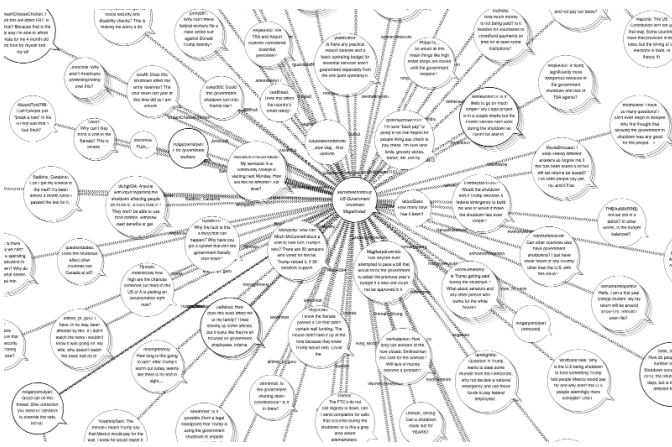


Fig. 9. Visualization of a Reddit megathread as a Distributed Noracle space.

the introduction of social learning bots. The diversity of participant needs suggested that this new facilitating tool should guide the users through the problem space, tailoring themselves to the user by analyzing the previously monitored usage data. Bots offered a seamless choice for providing this service on top of the existing application and framework.

#### F. Noracle Bot Requirement Analysis

To gather requirements on what type of information the Noracle Bot should report, we conducted a small study with five educational experts from the domain of our first pedagogical usage evaluation (cf. Sec. VI-D). While we do not see this study being an “own” iteration of our design science process, it provided us with the requirements that influenced the development of the pilot bot iteration described in the next section.

**Procedure:** In this study, we told the participants that we intended to create an analytical tool to help users engage with the Distributed Noracle application. We briefly explained the concept of a social bot, so that participants would understand how to frame their requirements. We then asked them to engage with the tool as part of a regular reflection activity and to observe the space, the users, and the content they create. Following the activity with the Distributed Noracle, we delivered a guided meditation, asking participants to consider what could happen if the number of questions in the graph were to increase ten-fold, or if more users were to participate. Finally, we asked them what they believed might happen if users were engaged in multiple spaces with different types of seed questions at the same time.

**Analysis and Outcomes:** Participants mentioned capabilities that the bot should have including informing you if someone interacted with one of your questions or other contributions in the space. In addition, participants were interested in seeing high quality information that helped them understand and navigate the space. For example, participants were interested in which questions provoked the most activity, agreement or disagreement in the space, as well as which question threads appeared most dense. Finally, the bot should provide both general and personalized reports related to different participants’ activity in the space.

#### G. Noracle Bot Pilot Evaluation

On the basis of this requirements analysis, we created the *Noracle Bot*, a social bot for the Distributed Noracle, and conducted a pilot evaluation to test both the capabilities of this first bot prototype, as well as our evaluation procedure for the upcoming second real-world pedagogical usage evaluation. Following recommendations from the first pedagogical usage evaluation, we facilitated a Distributed Noracle space for this pilot evaluation and also included participants who were familiar with the face-to-face or digital versions of the Distributed Noracle. In addition to facilitation by these researchers, these influencers’ role was to help shape the activities of the pilot “community”. Participants were invited to use the space as they might if they were actually working together on a topic, and to observe the functionality of the Noracle Bot as the question-graph expanded.

**Participants and Procedure:** 9 participants from the researchers’ networks of past participants and colleagues took part in the evaluation, with one participant being in a location with an internet firewall that rejected the application and thus not being able to participate. The participant group was diverse, with different nationalities, levels of experience, domains of activity, age and knowledge about the method of question-based dialog used in Distributed Noracle.

We provided a dummy seed question of “Do you have a question?” to allow participants to choose which themes they wanted to bring up in their questioning. After receiving instructions on how to participate in the Distributed Noracle space, the evaluation began. We also asked participants to use the tool for five days during the work week and to make at least three contributions per day.

At the end of the week, we prepared an anonymous online evaluation for participants and invited more in-depth feedback both in-person and via e-mail. The questions provided to participants in the evaluation were intended to refer back to the basic requirements that previous participants mentioned: to provide more information about one’s own activity in the space and to identify points of interest in the graph, delivered in a way that users find manageable.

**Analysis and Outcomes:** Three days into the evaluation, the question graph became so large that participation was very difficult. However, the evaluators attempted to continue until we stopped the evaluation exercise one day early. Six out of the eight participants who were able to participate in the evaluation had positive impressions of the Noracle Bot from the evaluation, rating the quality of information provided as “high” or “very high”. In addition, the majority of participants believed that the bot was most likely necessary for navigating the space effectively. With regard to the bot’s ability to provide information about one’s own activity in the space, seven participants described it as “valuable” (one participant offering a modifier of “highly valuable”). With regard to helpfulness in identifying points of interest, the bot performed slightly worse, with six participants describing the bot’s performance as “helpful” and both one modifier of “very helpful” and “neutral”. Concerning the delivery of messages, the diversity of participant answers indicated that in the future, the timing of

the bot's messages and the exact information it provides should be customizable. With the open questions that we provided in the evaluation to gather more qualitative data, we learned that the participants did evaluate the Noracle Bot as being mostly successful at the capabilities it was designed to do and that participants found those things useful in similar ways to those of the participants in the requirements evaluation. One particular interesting improvement mentioned was to have the "flexibility in when to call the stats and a deeper look at how many branches you created or how many child relations came from one of your questions."

#### H. Second Real-World Pedagogical Usage Evaluation

Our second real-world pedagogical usage evaluation took place in the context of an Austrian student association training event for young people that lasted several days. The trainers organizing the event were in a generational change, with some of the more experienced trainers on the edge of retirement and some novice trainers participating for the first time. Also the handover of the event management, which took place after this year's event, was announced well in advance. With this background, many discussions were held about the future direction of the event. Noracle was chosen as a method to support these discussions and open them up to all staff members. We defined *three main areas of discussion* and formulated open ended questions to initiate *three predefined Distributed Noracle spaces*. The evaluation participants had the goal to create a comprehensive, collective understanding of what their burning questions for the future of the event were.

**Participants and Procedure:** Evaluation participants were ten trainers from this event. The participants were male, aged between 20 and 30 years with different professions. The event consisted of lectures, discussions, practical activities, and evening events. Every trainer had a different schedule and different tasks during the day, such that face-to-face group reflection was only possible after 10pm. The event started on a Friday at 4pm. Scheduled event activities, debriefing and preparations for Saturday kept the trainers busy till midnight. After that, the Distributed Noracle was introduced as a tool to facilitate collaboration and group thinking, despite the different schedules. Technical support and guidance was provided to ensure that the system was working properly for all participants. They were instructed to use the Distributed Noracle for the coming days of the event, and that they would receive social bot messages regarding the activity of the question-based dialog via Slack. The social bot was configured to send out messages at 10pm such that participants would also be able to discuss the provided information. The evaluation ended on the next Wednesday's morning and participants were asked to fill a questionnaire.

**Analysis and Outcomes:** The participants started creating questions on Friday night. The last question was asked on Tuesday afternoon. Within this time frame 54 questions, linked with 58 relations, were asked and 22 votes were cast. Of those 54 questions, 34 were created between 8am and 10am, 10 between 4pm and 6pm. Between 10am and noon, as well as

between noon and 2pm, 3 questions were created each. Finally, 4 questions were created between midnight and 2am, all on the first night after the tool was introduced. This distribution of activity aligns with the expectations we had from the different schedules of the trainers. As the questions were asked over a period of about 85 hours, on average an average of 15 new questions were asked per day. Most activity was recorded on the second full day of activity (Sunday). The ratio between new relations and new questions rose from 0.75 new relations per new question on Saturday to 1.19 on Sunday and peaked at 1.38 on Monday before falling to 1 on Tuesday.

To cover the difference in perception between single-space and multi-space usage, we extended the questionnaire from the pilot evaluation, such that we could compare the perceived differences in terms of sense making of what is happening, understanding ones own activity and pointing to questions of potential interest. Eight questionnaires were handed in. With respect to the multi-space evaluation, participants perceived most value of "the Noracle Bot in pointing you to questions of potential interest" across spaces, as compared to a single space scenario. Two respondents perceived the Noracle Bot very valuable in this regard across multiple spaces, but only somewhat valuable for a single space. Another respondent perceived it extremely valuable across multiple spaces and very valuable for a single space. The other five respondents found it equally valuable. This observed difference in perception on single and multi-space scenarios is not significant, but well aligned with the intuition that the value of a bot raises with growing complexity.

From the open ended questions we extracted the following reoccurring themes from the responses (occurrences in brackets):

- Question-based dialog as method triggers thinking in new structures (6). It was noted however, that it may not suit every problem (1).
- Bot messages were informative (6) about structure (4), where questions belong and connections should be modeled (2).
- Information provided by the bot needs to be (more) precise/specific (3) and directly related to ones own questions or actions (2).

Responses to whether or not the participant would recommend using the Distributed Noracle to a friend or colleague, resulted in two detractors (those who would not recommend), one passives (who would not agree to recommending or not recommending) and five promoters (who would recommend).

#### I. Discussion

Improvements proposed by users mostly dealt with the interface and analytic features, such as additional ways of visualizing other aspects of the dialog by making nodes larger or smaller, allowing for certain questions to be marked as "resolved" and additional ways of linking questions. Most of the users in all evaluations said that such a tool can be useful in the planning stages of a project and at the beginning of any complex task or assignment to gain orientation. In addition, participants saw affordances for structuring group-

and teamwork in schools. Improvements suggested by the evaluators also included providing more information about the tool and how to use it, to make the graph searchable by keyword, and improving the interface.

From the technical point of view, the evaluations showed potential weak points of our application, such as the stability and ease of starting a node. While we were able to solve many technical challenges and improve the system during and after each of the evaluations, we are still working on improving both points. Nevertheless, the two real-world pedagogical usage evaluations proved that our prototype is applicable in real-world usage scenarios.

The trainer team of the first real-world pedagogical usage evaluation stated they were able to save considerable time in gathering important information on the trainees' expectations and knowledge. In a typical training scenario, a half day would have been spent on these types of abstract questions about the program. In this case, it only took 45 minutes of analyzing the resulting question-graph to achieve an even better result. In addition, starting the process in advance seemed to have the effect that the group took the exercise more seriously, which lead to these better results. Possible reasons for this mentioned by the trainers were that when the method is used in face-to-face settings, the participants are naturally distracted by the person they have in front of them. The tendency to move towards providing answers or advice makes it more difficult to keep them on task. Working asynchronously with the participants appeared to have resolved this as it was not necessary to always repeat that the participants should only ask questions.

Our second real-world pedagogical usage evaluation did show the helpfulness of the Noracle Bot, especially in a multi-space scenario. This is still a topic to be explored further, since we only scratched the surface of the possibilities social bots bear, and we did not evaluate the (potential) role a social bot plays in the community itself by being a real member of the space and e.g. creating questions, based on previous domain knowledge. These processes, supported by the rising trend of deep learning [29], bear a lot of potential for further investigation of user guidance and nudging in CoPs.

## VII. RELATED WORK

Question asking is seen as one of the most important skills for knowledge building, contributing to lateral thinking and better problem solving [30]. Question-based dialog is viewed as a specific type of a sense-making tool that is also *representation-centric* [31]. To help structure discourse analysis, computational linguistics has offered frameworks to examine collaborative sense-making in virtual environments [32]. For example, argumentation platforms offer a representation-centric approach to collaboration. Contributions are visually represented, categorized as issues, claims, premises and evidence, with modifying functions to support or refute other constituents of the argument. Cohesion graphs of discussion threads, which represent contributions as nodes at different levels, can examine lexical chains in discourse analysis to understand influence on conversation and identify key issues

in conversation. Related work in this domain mostly deals with the issue of how face-to-face scenarios differ from online discussions and how to aggregate community knowledge [33]. Instead of representing *knowledge* in the form of arguments, the Distributed Noracle examines the *gaps* in community knowledge in the form of questions.

The question of system maturity, flexibility and also interoperability is still an active research area [34]. The information infrastructure [35] of a p2p information system consists of the physical infrastructure, the p2p storage overlay network and the p2p service overlay network. The idea of using p2p-based information systems for sharing of educational resources came up first with the creation of EDUTELLA [36], a network for exchanging information about learning objects. Another active research domain is the use of p2p information systems for (decentralized) social networks (e.g. [37], [38]), as the p2p conceptual architecture is closer to the actual nature of communication and collaboration in online communities [39]. Further development in this area is driven by the InterPlanetary File System (IPFS) [40] project, which describes itself as a *peer-to-peer hypermedia protocol* and shares the concern for *increasing consolidation of control [on the Web]*. Related development approaches have been characterized as p2p cloud computing [41] and edge-centric computing [42].

Research of bots already started in the 1960s with the development of "Eliza" by Joseph Weizenbaum [43]. More recent research and especially developments of social bots in the domain of learning technologies starts around the year 2005, with Fryer and Carpenter using chatbot technology to support the acquisition of language skills [44]. These early works of modern bot development were quite limited and only able to reply to specific questions with predefined answers. Kerly et al. used a chatbot to support the self-assessment and reflection of learners [45], where learners were able to discover and negotiate their own learner model by using the chat tool. In order to bring the relevance of bots closer to students, they are also integrated into introductory courses for CS students [46]. A team from the University of Edinburgh used a chatbot for one of their courses [47]. The bot replied to Twitter tweets, thereby taking the role of a teacher. The replies were based on keywords in the students' request. As an example, the bot was able to answer questions regarding the submission deadline of assignments. The idea of this approach was to overcome possible shyness of students through conversation with a bot instead of a human teacher [48]. Krafft et al. use bots as virtual confederates, which behave like human confederates in experimental situations [49]. In their work they give four different ways to experiment on the bots. They propose to randomize bot actions, attributes, behavior and the artificial which is created by multiple bots [49]. Dibitonto et al. present the design of LiSA, a social bot for Facebook which conducts surveys with students [50]. The bot interviewed the students mainly about the way they obtained information about the university.

Despite the high research activity in the domains our contribution touches (decentralized infrastructures, CoP support, social bot utilization and support of community learning processes in general), we did not find any recent approaches



that provide a holistic support for CoPs with a self-managed, decentralized infrastructure. Forums, blogs and wikis are still the most commonly adopted tools for CoPs that need to accommodate geographically distributed participants at scale. However, they do not preserve a representation of contributions that can be elaborated or amended as the community changes, making them harder to sustain for CoPs.

### VIII. CONCLUSION

In this contribution, we presented both a decentralized learning infrastructure for distributed CoPs and an application developed with it in form of a question-based dialog tool for knowledge building. The infrastructure is based on p2p principles and can be managed by the community members themselves. The microservices realizing the applications build with the infrastructure self-replicate through the network according to the community's current needs and provide the necessary information. The developed service explorer makes it easy for community members to monitor and steer this process from the Web browser. The introduction of the Noracle Bot, as an exemplary utilization of conversational interfaces, connects the decentralized learning environment infrastructure to already established tool support in CoPs.

We followed a design science approach and incrementally tailored our application to the needs of the community, according to the outcome of each evaluation. Our real-world pedagogical usage evaluations proved the applicability of our approach in the domain of non-formal learning communities. The evaluations also showed that a digital version of a proven method for inquiry-based learning affects the community's knowledge of their ignorance (RQ 1) and can be build with our presented infrastructure, managed by the community itself (RQ 2). The Distributed Noracle provides both potential time-saving opportunities, as well as it enables question-based dialog sessions that would otherwise just not be applicable because of spacial differences within the CoP, or the lack of centralized, managed infrastructure.

Our approach concentrates on taking into account the specific attributes of CoPs, like temporal and spatial dynamics. By consequently addressing these attributes, we support CoPs in their efforts to share and acquire knowledge. As information remains available throughout the communities' existence and services evolve continuously at the same time, our infrastructure ensures sustainability and adaptability, aptitudes we reckon to be crucial in the development of a more democratic and egalitarian Web.

### REFERENCES

- [1] D. Schugurensky, *The Forms of Informal Learning: Towards a Conceptualization of the Field*, ser. NALL Working Paper. Centre for the Study of Education and Work, 2000.
- [2] H. Eshach, "Bridging In-School and Out-of-School Learning: Formal, Non-Formal, and Informal Education," *Journal of Science Education and Technology*, vol. 16, no. 2, pp. 171–190, 2007.
- [3] E. Wenger, *Communities of Practice: Learning, Meaning, and Identity*, ser. Learning in doing. Cambridge, UK: Cambridge University Press, 1998.
- [4] C. Anderson, *The Long Tail: Why the Future of Business Is Selling Less of More*. New York: Hyperion, 2006.
- [5] S. Newman, *Building Microservices: Designing Fine-Grained Systems*. Sebastopol, CA, USA: O'Reilly, 2015.
- [6] T. Farrell-Frey, G. Gkotsis, and A. Mikroyannidis, "Are you thinking what I'm thinking? Representing Metacognition with Question-based Dialogue," in *6th Workshop on Awareness and Reflection in Technology Enhanced Learning*, 2016.
- [7] M. Scardamalia and C. Bereiter, "Knowledge Building: Theory, Pedagogy, and Technology," *Cambridge Handbook of the Learning Sciences*, 2006.
- [8] I. Nonaka and H. Takeuchi, *The Knowledge-Creating Company*. Oxford University Press, 1995.
- [9] M. A. Chatti, R. Klamma, M. Jarke, and Ambjörn Naeve, "The Web 2.0 Driven SECI Model Based Learning Process," in *7th IEEE International Conference on Advanced Learning Technologies*. IEEE Computer Society, 2007, pp. 780–782.
- [10] M. Scardamalia, "Collective Cognitive Responsibility for the Advancement of Knowledge," *Liberal education in a knowledge society*, vol. 97, pp. 67–98, 2002.
- [11] A. Suarez, S. Ternier, M. Kalz, and M. Specht, "Supporting Inquiry-Based Learning with Google Glass (GPIM)," *Interaction Design and Architecture (s) Journal-IxD&A*, vol. 24, pp. 100–110, 2015.
- [12] C. Lotter, J. A. Yow, and T. T. Peters, "Building a Community of Practice Around Inquiry Instruction Through a Professional Development Program," *International Journal of Science and Mathematics Education*, vol. 12, no. 1, pp. 1–23, 2014.
- [13] A. R. Hevner, S. T. March, J. Park, and S. Ram, "Design Science in Information Systems Research," *MIS Quarterly*, vol. 28, no. 1, pp. 75–105, 2004.
- [14] K. Peffers, T. Tuunanen, M. A. Rothenberger, and S. Chatterjee, "A Design Science Research Methodology for Information Systems Research," *Journal of Management Information Systems*, vol. 24, no. 3, pp. 45–77, 2007.
- [15] P. de Lange, T. Farrell-Frey, B. Göschlberger, and R. Klamma, "Transferring a Question-Based Dialog Framework to a Distributed Architecture," in *Data Driven Approaches in Digital Education- 12th European Conference on Technology Enhanced Learning*, ser. Lecture Notes in Computer Science, vol. 10474. Springer International Publishing, 2017, pp. 549–552.
- [16] P. de Lange, B. Göschlberger, T. Farrell, and R. Klamma, "A Microservice Infrastructure for Distributed Communities of Practice," in *Lifelong Technology-Enhanced Learning - 13th European Conference on Technology Enhanced Learning*, ser. Lecture Notes in Computer Science, vol. 11082. Springer International Publishing, 2018, pp. 172–186.
- [17] P. de Lange, T. Janson, and R. Klamma, "Decentralized Service Registry and Discovery in P2P Networks Using Blockchain Technology," in *Web Engineering*, ser. Lecture Notes in Computer Science. Springer International Publishing, 2019.
- [18] R. Klamma, D. Renzel, P. de Lange, and H. Janßen, "las2peer – A Primer," ResearchGate, 2016.
- [19] A. Rowstron and P. Druschel, "Pastry: Scalable, Decentralized Object Location, and Routing for Large-Scale Peer-to-Peer Systems," in *Middleware 2001: IFIP/ACM International Conference on Distributed Systems Platforms*, ser. Lecture Notes in Computer Science. Springer, 2001, pp. 329–350.
- [20] R. Fielding and R. Taylor, *Architectural Styles and the Design of Network-Based Software Architectures*. University of California, Irvine Doctoral Dissertation, 2000.
- [21] G. Banavar, T. Chandra, R. Strom, and D. Sturman, "A Case for Message Oriented Middleware," in *International Symposium on Distributed Computing*. Springer, 1999, pp. 1–17.
- [22] P. Eugster, P. Felber, R. Guerraoui, and A.-M. Kermarrec, "The many Faces of Publish/Subscribe," *ACM Computing Surveys (CSUR)*, vol. 35, no. 2, pp. 114–131, 2003.
- [23] M. Richards, *Microservices vs. Service-Oriented Architecture*. O'Reilly Media, 2015.
- [24] D. Pritchett, "BASE: An ACID Alternative," *Queue*, vol. 6, no. 3, pp. 48–55, 2008.
- [25] G. Wood, "Ethereum: A Secure Decentralized Transaction Ledger," 2014.
- [26] M. Kisters and J. van der Heijden, "From Mechanism to Virtue: Evaluating Nudge Theory," *Evaluation*, vol. 21, no. 3, pp. 276–291, 2015.
- [27] D. Renzel, M. Behrendt, R. Klamma, and M. Jarke, "Requirements Bazaar: Social Requirements Engineering for Community-Driven Innovation," in *21st IEEE International Requirements Engineering Conference*, 2013, pp. 326–327.

- [28] D. Renzel, R. Klamma, and M. Jarke, "IS Success Awareness in Community-Oriented Design Science Research," in *New Horizons in Design Science: Broadening the Research Agenda*, ser. Lecture Notes in Computer Science, vol. 9073. Springer International Publishing, 2015, pp. 413–420.
- [29] Deng, Li and Yu, Dong, "Deep Learning: Methods and Applications," *Foundations and Trends in Signal Processing*, vol. 7, no. 3–4, pp. 197–387, 2014.
- [30] P. Sloane, *The Leader's Guide to Lateral Thinking Skills: Unlock the Creativity and Innovation in You and Your Team*. Kogan Page, 2017.
- [31] C. McLoughlin, K. D. Patel, T. O'Callaghan, and S. Reeves, "The Use of Virtual Communities of Practice to Improve Interprofessional Collaboration and Education: Findings from an Integrated Review," *Journal of Interprofessional Care*, vol. 32, no. 2, pp. 136–142, 2018.
- [32] L. Iandoli, I. Quinto, A. de Liddo, and S. Buckingham Shum, "On Online Collaboration and Construction of Shared Knowledge: Assessing Mediation Capability in Computer Supported Argument Visualization Tools," *Journal of the Association for Information Science and Technology*, vol. 67, no. 5, pp. 1052–1067, 2016.
- [33] K. A. Meyer, "Face-to-Face Versus Threaded Discussions: The Role of Time and Higher-Order Thinking," *JALN*, vol. 7, no. 3, pp. 55–65, 2003.
- [34] X. Ochoa and S. Ternier, "Technical Learning Infrastructure, Interoperability and Standards," in *Technology Enhanced Learning*. Springer, 2017, pp. 145–155.
- [35] G. C. Bowker, K. Baker, F. Millerand, and D. Ribes, "Toward Information Infrastructure Studies: Ways of Knowing in a Networked Environment," in *International Handbook of Internet Research*. Dordrecht: Springer Netherlands, 2010, pp. 97–117.
- [36] W. Nejdl, B. Wolf, C. Qu, S. Decker, M. Sintek, A. Naeve, M. Nilsson, M. Palmér, and T. Risch, "EDUTELLA: a P2P Networking Infrastructure Based on RDF," in *Proceedings of the 11th International World Wide Web Conference*, ser. WWW '02. ACM, 2002, pp. 604–615.
- [37] S. Nilizadeh, S. Jahid, P. Mittal, N. Borisov, and A. Kapadia, "Cachet: A Decentralized Architecture for Privacy Preserving Social Networking with Caching," in *8th International Conference on Emerging Networking Experiments and Technologies*, ser. CoNEXT '12. New York, NY, USA: ACM, 2012, pp. 337–348.
- [38] B. Guidi, T. Amft, A. De Salve, K. Graffi, and L. Ricci, "DiDuSoNet: A P2P Architecture for Distributed Dunbar-Based Social Networks," *Peer-to-Peer Networking and Applications*, vol. 9, no. 6, pp. 1177–1194, 2016.
- [39] S. Buchegger and A. Datta, "A Case for P2P Infrastructure for Social Networks - Opportunities & Challenges," in *Sixth International Conference on Wireless On-Demand Network Systems and Services*, 2009, pp. 161–168.
- [40] J. Benet. (2014) IPFS - Content Addressed, Versioned, P2P File System. Protocol Labs.
- [41] O. Babaoglu and M. Marzolla, "The People's Cloud," *IEEE Spectrum*, vol. 51, no. 10, pp. 50–55, 2014.
- [42] P. Garcia Lopez, A. Montresor, D. Epema, A. Datta, T. Higashino, A. Iamnitchi, M. Barcellos, P. Felber, and E. Riviere, "Edge-Centric Computing: Vision and Challenges," *ACM SIGCOMM Computer Communication Review*, vol. 45, no. 5, pp. 37–42, 2015.
- [43] J. Weizenbaum, "ELIZA—A Computer Program for the Study of Natural Language Communication Between Man and Machine," *Communications of the ACM*, vol. 9, no. 1, pp. 36–45, 1966.
- [44] L. Fryer and R. Carpenter, "Bots as Language Learning Tools," *Language Learning & Technology*, vol. 10, 2006.
- [45] A. Kerly, P. Hall, and S. Bull, "Bringing Chatbots into Education: Towards Natural Language Negotiation of Open Learner Models," *Knowledge-Based Systems*, vol. 20, no. 2, pp. 177–185, 2007.
- [46] A. Shaw, "Using Chatbots to Teach Socially Intelligent Computing Principles in Introductory Computer Science Courses," in *2012 Ninth International Conference on Information Technology - New Generations*. IEEE, 2012, pp. 850–851.
- [47] S. Bayne, "Teacherbot: Interventions in Automated Teaching," *Teaching in Higher Education*, vol. 20, no. 4, pp. 455–467, 2015.
- [48] A. Bozkurt, W. Kilgore, and M. Crosslin, "Bot-Teachers in Hybrid Massive Open Online Courses (MOOCs): A Post-Humanist Experience," *Australasian Journal of Educational Technology*, 2018.
- [49] P. M. Krafft, M. Macy, and A. S. Pentland, "Bots as Virtual Confederates," in *20th ACM Conference on Computer-Supported Cooperative Work and Social Computing*. ACM Press, 2017, pp. 183–190.
- [50] M. Dibitonto, K. Leszczynska, F. Tazzi, and C. M. Medaglia, "Chatbot in a Campus Environment: Design of LiSA, a Virtual Assistant to Help Students in Their University Life," in *Human-Computer Interaction. Interaction Technologies*, ser. Lecture Notes in Computer Science. Springer International Publishing, 2018, vol. 10903, pp. 103–116.



and peer to peer infrastructures.



**Peter de Lange** received his Master's degree in computer science in 2015 from RWTH Aachen University. Peter works as a research assistant and PhD student within the Advanced Community Information Systems group at the Information Systems and Databases chair, RWTH Aachen University, Germany since 2015. He has been involved in both German and European projects in Technology Enhanced Learning as well as research and teaching. Among his research interests are Web engineering, community information systems, learning analytics

**Bernhard Göschlberger** received his Master's degree in Software Engineering from the University of Applied Sciences Upper Austria. Bernhard is Deputy Studio Director of the Research Studio Data Science and team leader of the MobileKnowledgeLab at the Research Studios Austria Forschungsgesellschaft. He is PhD student at the Institute of Telecooperation at the Johannes Kepler University Linz.

**Tracie Farrell** is a research associate working with the Web Science research group at the Knowledge Media Institute (KMI) of the Open University UK. Her current research interests include the impact of new technologies on society and culture, in particular, the opportunities and challenges presented by communication and interaction at scale.

**Alexander Tobias Neumann** received his master's degree in Computer Science in 2018 from RWTH Aachen University. Alex works as a research assistant and PhD student within the Advanced Community Information Systems group at the Information Systems and Databases chair, RWTH Aachen University, Germany since 2018. His research focus lies on scaling up digital mentoring processes.

**Ralf Klamma** holds diploma, doctoral and habilitation degrees in computer science from RWTH Aachen University. He leads the research group "Advanced Community Information Systems" (ACIS) at the information systems chair, RWTH Aachen University. He is known for his work in major EU projects for Technology Enhanced Learning (PROLEARN, GALA, ROLE, Learning Layers, TELMAP, Tellnet, CUELC, SAGE, BOOST, VIRTUS and WEKIT). Ralf organized doctoral summer schools & conferences in Technology Enhanced Learning and Social Network Analysis. He published more than 150 scientific papers. He is on the editorial board of Social Network Analysis and Mining (SNAM) and IxD&A. His research interests are community information systems, serious games, augmented reality, web engineering, social network analysis, requirements engineering and technology enhanced learning.